SecureDoc™ DISK ENCRYPTION

for Linux

WINMAGIC DATA SECURITY | Knowing You're Protected

SecureDoc™ Linux 4.91-3, February 2010

©Copyright 1997 - 2010 by WinMagic Inc.

All rights reserved.

Printed in Canada

Many products, software and technologies are subject to export control for both Canada and the United States of America. WinMagic advises all customers that they are responsible for familiarizing themselves with these regulations. Exports and re-exports of WinMagic Inc. products are subject to Canadian and US export controls administered by the Canadian Border Services Agency (CBSA) and the Commerce Department's Bureau of Industry and Security (BIS). For more information, visit WinMagic's web site or the web site of the appropriate agency.

WinMagic, SecureDoc, SecureDoc Enterprise Server, Compartmental SecureDoc, SecureDoc PDA, SecureDoc Personal Edition, SecureDoc RME, SecureDoc Removable Media Encryption, SecureDoc Media Viewer, SecureDoc Express, SecureDoc for Mac, MySecureDoc, MySecureDoc Personal Edition Plus, MySecureDoc Media, and SecureDoc Central Database are trademarks and registered trademarks of WinMagic Inc., registered in the US and other countries. All other registered and unregistered trademarks herein are the sole property of their respective owners. © 2009 WinMagic Inc. All rights reserved.

Acknowledgements

This product includes cryptographic software written by Antoon Bosselaers, Hans Dobbertin, Bart Preneel, Eric Young (eay@mincom.oz.au) and Joan Daemen and Vincent Rijmen, creators of the Rijndael AES algorithm.

"This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.OpenSSL.org/)"

WinMagic would like to thank these developers for their software contributions.

Contacting WinMagic

WinMagic
200 Matheson Blvd West, Suite 201
Mississauga, Ontario, L5R 3L7

toll free: 1-888-879-5879
phone: (905) 502-7000
fax: (905) 502-7001

Sales:	sales@winmagic.com
Marketing:	marketing@winmagic.com
Human Resources:	hr@winmagic.com
Technical Support:	support@winmagic.com
Information:	info@winmagic.com
Billing inquiries:	finance@winmagic.com

Table of Contents

Chapter 1: About SecureDoc Linux 1
About SecureDoc Linux 1
About Full Disk Encryption 1
System Requirements 3
Limitations 3
License Agreement 3

Chapter 2: Installing SecureDoc Linux 4
Installing Managed SecureDoc Linux 4
Installing SecureDoc Linux Standalone 6
Overview 6
Installing SecureDoc Linux Package 6
Installing Boot Logon 7
Verify Kernel Module Installed 8
Installing/Updating Kernel Module 8

Chapter 3: Using SecureDoc Linux 10
Encrypting the Hard Disk 10
Uninstalling SecureDoc Linux 11
Overview 12
Decrypting the Hard Disk 12
Uninstalling Kernel Module 13
Uninstalling Service (for SES Managed Installs only) 13
Restoring MBR 13
Uninstalling SecureDoc Linux Package 14
Upgrading SecureDoc Linux 15
Upgrading the RPM package 15
Upgrading the Manual Installation Package 15
Updating Keyfiles (Standalone Only) 16
Listing Key Files in the System 16
Exporting a Key File 17
Importing a Key File 17
Deleting a Key File 17
Using Encrypted Removable Media 18
Mounting a USB Stick 18
Unmounting a USB Stick 18

Emergency Disk 19
Removing BootLogon 19
Restoring SecureDoc Space 20

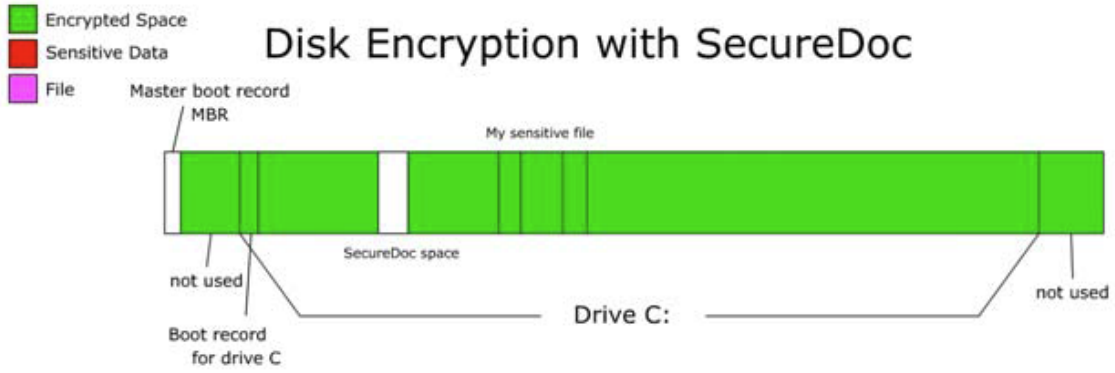
Chapter 3: Reference 22
Directory Structure 22
Interpreting Log Files 23
If Installation Check Fails 23
If No Log File 23
If Log File Contains Evident Errors 23
If Log File Contains No Evident Errors 23

Chapter 1: About SecureDoc Linux

About SecureDoc Linux

SecureDoc Linux is a standalone product to perform Full Disk Encryption (FDE) of the entire system hard disk. SecureDoc Linux supports centralized deployment through SecureDoc Enterprise Server (SES) as well as Standalone. The standalone version supports dual boot with Windows and Linux.

About Full Disk Encryption



Full disk encryption encrypts all data on sector-addressable storage media It encrypts the entire storage media in a single pass, during an initial phase called “conversion”. Once conversion is complete, subsequent encryption and decryption operations are transparent to users. Data is transparently intercepted and encrypted just before it is written to the disk, and intercepted and decrypted immediately after it is read from the disk. Interception and encryption / decryption occur at the point of sector-level disk access.

If a file from a fully encrypted disk is saved elsewhere other than the encrypted disk, it remains in plain text. For example, if a file is opened and saved to a network folder, the file remains in plain text on the network, as the file has not been re-encrypted back to the hard disk.

The principal benefit of full disk encryption is more comprehensive protection for data-at-rest. Full disk encryption protects every file and all data saved to disk, including the operating system, executable files and users' documents. Disk encryption also protects temporary, recycled, and paging files. No other method can thoroughly protect all of these files as well as data not addressable as a file.

It is important to note that data, once written to magnetic media such as a hard disk, can be recovered even after it has been overwritten. Once conversion is completed, data is never written to the media in plain text form.

Unauthorized users cannot read any data, even the file name, file size, or folder structure.

Full disk encryption is widely regarded as the best practice for ensuring the confidentiality of PII and proprietary digital assets stored on mobile devices and removable media.

System Requirements

> **IMPORTANT:**
> Most Linux distributions have a software update capability that can update the kernel running on the system. If you perform a kernel update after SecureDoc Linux is installed and the drive is encrypted, you MUST update SecureDoc before rebooting. **Failure to do so can result in a non-bootable system.** See "Installing/Updating Kernel Module" on page 8.

For a standalone install, you must have access to a valid encryption Keyfile that was created by SecureDoc (on a Windows machine with SecureDoc or SES and copied to this computer): you need to know the KeyFile default password and the name of at least one key in the keyfile.

SecureDoc Linux is available for the following Linux distributions:

- SUSE Linux Enterprise Desktop 11
- OpenSUSE 10.2, 11.0, 11.1
- RedHat Enterprise Linx (RHEL) Server 5.3 & 5.4
- RedHat Enterprise Linx (RHEL) Desktop 5.3
- Fedora 10, 11
- Debian 5.0

SecureDoc only supports 32-bit on an Intel processor. To check your machine's processor, enter `uname -m` and ensure the processor is at least .i586 and does not contain _64.

If the system uses LVM style partitioning then it must have a /boot partition, otherwise it must have a swap partition.

Limitations

SecureDoc Linux does not currently support the following features found in SecureDoc for other platforms:

- Hardware tokens
- Removable media including CD, DVD. SecureDoc Linux can read encrypted USB media from another platform, provided you have the proper encryption key, but cannot encrypt USB media itself.

License Agreement

If you use this software you are bound by the legal agreements in the license agreements file located in /usr/local/WinMagic/share.

Chapter 2: Installing SecureDoc Linux

Installing Managed SecureDoc Linux

Note: Managed SecureDoc Linux installs do not currently support dual boot environments.

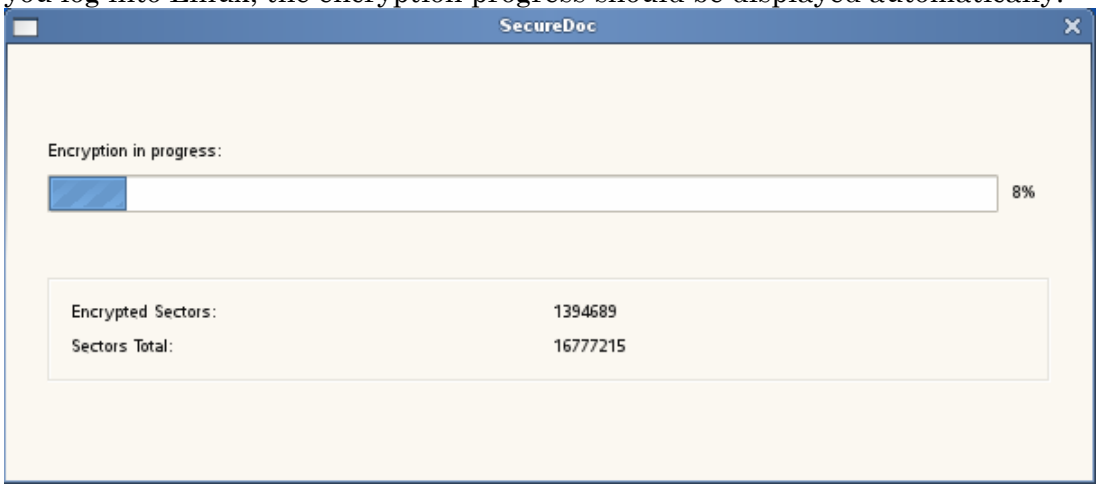
Your SES Administrator should have provided you with the following installation files:

- `Boot_msg.txt`
- `PackageSettings.ini`
- `SDConnex.cer`
- `SDProfile.spf`
- `wm_install`
- `wm_secdoc.rpm`

Install a Managed SecureDoc Linux as follows:

1. Open a terminal window.
2. Switch user to root.
3. Copy the installation files listed above to an appropriate location.
4. Change to the directory where the files were copied
   ```
   # cd {path}
   ```
5. Run
   ```
   # ./wm_install
   ```
   You should see the following messages
   ```
   Checking dependencies... OK
   Installing... Creating application symbolic links... OK
   SDService installing...
   Connecting to SES... [OK]
   Registering computer/user... [OK]
   Boot logon installing...
     System uses LVM style partitioning.
     Resizing boot partition...
   Boot logon installed successfully
   Installing service...
   Service installed successfully
   Installing kernel module...
   Kernel module installed successfully
   You must reboot your machine
   Reboot computer now (y/n)?
   ```

6. If successful, you will be prompted to reboot: press “y” ENTER.
7. When the computer restarts you will be displayed with Boot Logon and required to enter the initial password provided by your SES Administrator. Once you do, Linux should boot normally
8. Once Linux has booted, the drive will begin to be encrypted automatically. Once you log into Linux, the encryption progress should be displayed automatically.



NOTE: To view the progress of the encryption manually, run `/usr/local/WinMagic/bin/SDCCLin &`
You may continue to work on your computer while the encryption is underway.

9. When the encryption is complete, your system is protected.

Installing SecureDoc Linux Standalone

Overview

To install SecureDoc Linux:	To install SecureDoc Linux for a dual boot Windows and Linux system:
1. Install SecureDoc Package. 2. Install Boot Logon (which also installs the kernel module) and reboot. 3. Verify Kernel Module Installed. 4. Encrypt the Hard Disk.	1. Install SecureDoc Linux Package. 2. Install Kernel Module and reboot. 3. Verify Kernel Module Installed. 4. Reboot to Windows. 5. Install SecureDoc for Windows following the instructions in the SecureDoc Windows User Manual to create a key, install Boot Logon and encrypt the hard disk.

NOTE: Take care with the syntax surrounding "-". Enter commands exactly as they appear in this documentation.
All commands must be performed as root user.

Installing SecureDoc Linux Package

NOTE: You should create an image of the hard disk before installation. This allows you to restore the disk to its original state if necessary.

There are two different distributions of SecureDoc Linux: as an RPM for the majority of Linux distributions that support RPM, and as a tar file for manual installation on distributions that do not support RPM (e.g. Debian).

If your Linux supports RPM, install SecureDoc as follows:

1. Copy the installation package (e.g. `wm_secdoc-4.91-1.rpm`) to an appropriate location.
2. Enter the following in the Linux Terminal:
   `# rpm -i location/package`
   where `location` is where the installation package resides and `package` is the name of the package.
3. During installation, the package checks for the `parted` package before doing any file installation. If installation is successful, you see:

```
# rpm -i /tmp/wm_secdoc-4.91-1.rpm
Checking dependencies... OK
Installing... OK
Creating application symbolic links... OK
```

If installation is not successful, you see:

```
# rpm -i /tmp/wm_secdoc-4.91-1.rpm
Checking dependencies... FAILED
Check manually: rpm -qa | grep parted
Install parted package if necessary
```

4. Installing the RPM package copies all the necessary files to `/usr/local/WinMagic`.
5. To test package installation, enter:

```
# rpm -qa | grep wm_secdoc
wm_secdoc-4.91-1.i586
```

If your Linux does not support RPM, install SecureDoc as follows:

1. Copy the manual installation package (e.g. `wm_secdoc-4.91-1.manual.tar`) to `/usr/local`.
2. Enter the following in the Linux Terminal:

   `# cd /usr/local`

   `# tar –xvf` *`package`*

   where package is the name of the manual installation package.

NOTE: The tar must be extracted so that the path is /usr/local/WinMagic/…

Installing Boot Logon

NOTE: This process will require a reboot.

---

1. Copy your keyfile to the Linux machine.
2. Enter the following in the Linux terminal:

   `# wm_bootinstall --dbk=`*`path/keyfile`*`.dbk`

   Where *`path`* is the path to the keyfile and *`keyfile`* is the keyfile name. This will try and determine the primary boot drive for your system, typically `/dev/sda` or `/dev/hda`. You are prompted to confirm the target disk for installation.

   During installation, you can monitor its progress in another shell prompt:

   `# tail -f /usr/local/WinMagic/var/boot.log`

   For more on this log file, see “Interpreting Log Files” on page 23.
3. If the installation is successful, you should see the following lines at the bottom of the output:

```
Kernel module installed successfully
You must reboot your machine
```

   To reboot. Enter:

   `# reboot`

4. If the installation fails, the most common reason is that there is no suitable module found for your kernel version: contact SecureDoc for a patch (see "Installing/Updating Kernel Module" on page 8).
5. When the machine reboots to BootLogon, choose the default keyfile by pressing ENTER or entering "1". Then enter the password for the keyfile and press ENTER.

Verify Kernel Module Installed

1. When Linux restarts, verify the installation. Enter:
   ```
   # lsmod | grep wm_secdoc
   ```
   It should return:
   ```
   wm_secdoc             1830492  1
   ```
   If it does not, see "If Installation Check Fails" on page 23.
2. Enter:
   ```
   # ls -la /dev/wm_secdoc
   ```
   It should return:
   ```
   crw-r--r-- 1 root root 254, 0 2008-03-06 10:05 /dev/wm_secdoc
   ```
   These checks indicate that the module is loaded and the associated device link was created correctly.
3. If you are not installing on a dual boot system, backup (copy not move) the files in `/usr/local/WinMagic/var` to a secure location located off of the machine that you are working on.

Installing/Updating Kernel Module

IMPORTANT:
Most Linux distributions have a software update capability that can update the kernel running on the system. If you perform a kernel update after SecureDoc Linux is installed and the drive is encrypted, you MUST update SecureDoc before rebooting. **Failure to do so can result in a non-bootable system.**

Most Linux distributions that allow kernel updates will create a boot menu in GRUB with the old kernel and the new kernel so that if there is a problem booting the new kernel, you can reboot and select the old kernel from the boot menu. If you are unsure of what your Linux distribution does, contact your system administrator.

This process is done automatically after installing Boot Logon but may need to be done manually for dual boot with Windows or if a new kernel module update is required. You can update the kernel module when one is already installed but you will notice the message "ERROR: Module wm_secdoc is in use". You can ignore this message.

This scenario can be done with a plaintext or encrypted disk.

NOTE: For an encrypted disk, the machine cannot be rebooted until a new module is installed or else a non-bootable system may result.

1. Enter:

   ```
   # ls /usr/local/WinMagic/lib/wm_secdoc.ko-`uname -r`-`uname –m`
   ```

   If a file is listed then a kernel module already exists for your platform, otherwise enter:

   ```
   # uname -a
   ```

   Send the output of the above command to WinMagic Support. If one is available, WinMagic support will send you a new wm_secdoc.ko file. Copy it into the `/usr/local/WinMagic/lib` directory.

2. To install the kernel module on the currently running kernel enter:

   ```
   # /usr/local/WinMagic/bin/wm_moduleinstall
   ```

   To install the kernel module another kernel enter:

   ```
   # /usr/local/WinMagic/bin/wm_moduleinstall --kernelver={kernver}
   ```

   Where kernver is of the format returned from uname –r. To see what kernels are on your system, enter `ls /lib/modules`.

   Read all the outputs to spot any errors. The output is dependent of `mkinitrd` and `depmod` outputs and can be different from one Linux distribution to another.

3. If all goes well you should see the following at the end of the output:

   ```
   Kernel module installed successfully
   You must reboot your machine
   ```

Chapter 3: Using SecureDoc Linux

Encrypting the Hard Disk

1. Enter:
   `# wm_encrypt --key=KeyID`

   Where *KeyID* is the name of a key in the keyfile used at Boot Logon. If the key has spaces in its name, use quotation marks around the name (e.g., "first key").

   NOTE: In Windows, the key is prefixed with "AES". In Linux, the "AES" prefix is unnecessary.

2. You are prompted to confirm the encryption process:

   ```
   20080312141232 Encryption started
   Encrypt disk: /dev/hda (yes/no)?
   ```

   If the conversion is interrupted, the process is resumed using information from recovery files.

3. The encryption process is shown. For example:

   ```
   # wm_encrypt --disk=/dev/hda --key='1'
   20080312141649 Encryption started
   sector:     159745, percent: 0.95, epoch:   1205345840
   ```

   If an error occurs, the name of the log file is shown in the resulting message. For example:

   ```
   # wm_encrypt --disk=/dev/hda --key='1'
   20080312141553 Encryption started
   20080312141553 ERROR: encryption returns error
   ```

   Check the `/usr/local/WinMagic/var/encrypt.log` file for details and be prepared to send the log file to WinMagic Technical Support.

4. When encryption completes you should see:

   ```
   20080312141553 Encrypted successfully
   ************************************************************
   You should make a new backup copy of the files in
   /usr/local/WinMagic/var to some external media.
   ************************************************************
   ```

5. Make a backup copy of the files in `/usr/local/WinMagic/var` to a secure location off of the machine you are working on.

6. To check the log file, enter:
   `# less /usr/local/WinMagic/var/encrypt.log`

At any point you can check on the encryption state of the hard disk by entering:

```
# wm_diskstatus
```

Which should return one of the following values:

PLAINTEXT_MEDIA	Disk is not encrypted
PLAINTEXT_CHANGING	Disk is encrypting
ENCRYPTED_MEDIA	Disk is encrypted
ENCRYPTED_CHANGING	Disk is decrypting

Changing Password

To change your password, run:

```
# /usr/local/WinMagic/bin/SDCCLin [-password]
```

If you run SDCCLin without any command line parameters it will display the encryption progress if encryption/decryption is underway, otherwise it will display the change password prompt. You can force the change password prompt by specifying –password.

Uninstalling SecureDoc Linux

Overview

To uninstall SecureDoc Linux:	To uninstall SecureDoc Linux with a dual boot Windows and Linux system:
1. Decrypt the Hard Disk. 2. Uninstall Kernel Module and service (SES Managed Installs only), and reboot. 3. Restore MBR and reboot. 4. Uninstall SecureDoc Package.	1. Boot to Windows and uninstall SecureDoc for Windows following the instructions in the SecureDoc Windows User Manual to decrypt the hard disk, uninstall boot logon and uninstall the product. 2. Reboot to Linux. 3. Uninstall Kernel Module and reboot. 4. Uninstall SecureDoc Package.

Decrypting the Hard Disk

1. Enter:

   `# wm_decrypt --key=`*`Keyid`*

   Where *`Keyid`* is the name of a key in the keyfile used at Boot Logon.

2. You are prompted to confirm the decryption process:

   ```
   # 20080312141232 Decryption started
   Decrypt disk: /dev/hda (yes/no)?
   ```

3. Errors with the decryption are written to the decrypt.log file. For more on this log file, see “Interpreting Log Files” on page 23.

4. When decryption completes you should see:

   ```
   20080312141553 Decrypted successfully
   ************************************************************
   You should make a new backup copy of the files in
   /usr/local/WinMagic/var to some external media.
   ************************************************************
   ```

5. Make a backup copy of the files in `/usr/local/WinMagic/var` to a secure location off of the machine you are working on.

6. Check the status of the disk:

   `#wm_diskstatus`

   The result should indicate the disk is in PLAINTEXT_MEDIA format.

Uninstalling Kernel Module

To uninstall the kernel module from the running kernel, enter:

```
# /usr/local/WinMagic/bin/wm_moduleuninstall
```

If the disk is in plain text, no warnings are shown, you should see the following output.

```
20090414120307 Kernel module uninstalled successfully
20090414120307 You must reboot your machine
```

If the disk is not plain text, you are warned of this:

```
# /usr/local/WinMagic/bin/wm_moduleuninstall
WARNING: Disk status is: PLAINTEXT_CHANGING
WARNING: If you uninstall the kernel module the machine can
become unusable!
Uninstall the module (yes/no)? no
```

To uninstall the kernel module from another kernel, enter:

```
# /usr/local/WinMagic/bin/wm_moduleuninstall --kernelver={kernver}
```

Where kernver is of the format returned from uname –r. To see what kernels are on your system, enter `ls /lib/modules`. You should uninstall the kernel module for all versions of the kernel you installed it in.

Uninstalling Service (for SES Managed Installs only)

If this is an SES managed install, to uninstall the Service:

```
# /usr/local/WinMagic/bin/wm_serviceuninstall
Uninstalling service...
Service uninstalled successfully
```

Reboot the machine.

Restoring MBR

1. Enter
   `# /usr/local/WinMagic/bin/wm_mbrestore`
2. This will replace the MBR with the one that was saved MBR in the /usr/local/WinMagic/var directory during installation of SecureDoc Linux. This will try and determine the primary boot drive for your system, typically `/dev/sda` or `/dev/hda`. You are prompted to confirm:

```
# /usr/local/WinMagic/bin/wm_mbrestore
20080312153758 MBR sector restoring...
Overwrite MBR on disk: /dev/hda (yes/no)? yes
1+0 records in
1+0 records out
512 bytes (512 B) copied, 0.000147201 s, 3.5 MB/s
20080312153846 MBR sector successfully restored
```

3. To force the disk and the MBR file, use:

   ```
   #/usr/local/WinMagic/bin/wm_mbrestore --disk=/dev/{disk_device} --mbr=/path/mbr.pre.{timestamp}
   ```

   The previous command can be used to recover from disaster when you saved the `mbr.*` dump files from `/usr/local/WinMagic/var` after installation.

4. Reboot to ensure that Boot Logon has been removed.

Uninstalling SecureDoc Linux Package

To uninstall the RPM package:

1. Verify the package that is installed by entering:

   ```
   # rpm -qa | grep wm_secdoc
   wm_secdoc-4.91-1
   ```

2. Uninstall the package by entering:

   ```
   # rpm -e package_name
   ```

   Where *package_name* is the name of the package above, for example, `"wm_secdoc-4.91-1"`.

   The uninstall process checks for disk status if the kernel module is loaded. If disk status is anything but PLAINTEXT_MEDIA, you see the following error:

   ```
   Disk status is STATUS
   ```

   To force uninstall use the `--nopreun` parameter to the `rpm` command where STATUS is the status.

   NOTE: Force uninstall only when this error does not occur. Forcing uninstall when the disk is PLAINTEXT_CHANGING, ENCRYPTED_MEDIA or ENCRYPTED_CHANGING status will render the Linux root partition inaccessible.

3. The uninstall process will delete the package directory, delete the symbolic links. The output for a normal uninstall looks like this:

   ```
   Cleaning directory structure... OK
   Uninstalling the kernel module...
   ...
   ```

4. To ensure the package has been successfully removed, enter:

   ```
   # rpm -qa | grep wm_secdoc
   ```

   Nothing should be returned.

To uninstall the manual installation package:

1. Ensure you have performed all the previous uninstall steps. If you remove the installation directory when the product is still installed, you may end up with an inaccessible system.
2. Enter the following in the Linux Terminal:
   ```
   # cd /usr/local
   # rm -f WinMagic
   ```

Upgrading SecureDoc Linux

If you have previously installed SecureDoc Linux, you may use the following process to upgrade to the latest version.

Upgrading the RPM package

1. Copy the installation package (e.g. `wm_secdoc-5.0-1.rpm`) to an appropriate location.
2. Enter the following in the Linux Terminal:
   ```
   # rpm -U location/package
   ```
   where `location` is where the installation package resides and `package` is the name of the package.

Upgrading the Manual Installation Package

1. Copy the manual installation package (e.g. wm_secdoc-5.0-1.manual.tar) to /usr/local.
2. Enter the following in the Linux Terminal:
   ```
   # cd /usr/local
   # tar –xvf package
   ```
   where `package` is the name of the manual installation package.
3. Run
   ```
   # /usr/local/WinMagic/bin/wm_upgrade
   ```
4. Follow the instructions in section “Installing/Updating Kernel Module” on page 8 to update each of the kernels you are running.

Updating Keyfiles (Standalone Only)

Using the following procedures you can list, import, export and delete keyfiles from the SecureDoc Linux system if installed Standalone.

WARNING:

If you are running an SES-Managed SecureDoc Linux, you should not use this method to modify keyfiles.

You may want to do this if you need to:

- Change your password on your keyfile
- Add another encryption key to your keyfile, say to access some removable media
- Add another keyfile for an administrator to be able to log into your computer.

In all of the following commands you need to know the major and minor number for the HD. To determine this, run:

```
ls –l /dev/sd*  (on some systems it is "hd*")
brw-r----- 1 root disk 8,  0 2009-04-17 06:44 /dev/sda
brw-r----- 1 root disk 8,  1 2009-04-17 06:44 /dev/sda1
brw-r----- 1 root disk 8,  2 2009-04-17 06:44 /dev/sda2
brw-r----- 1 root disk 8,  3 2009-04-17 10:44 /dev/sda3
```

Note the major number 8 and minor number 0 for the HD /dev/sda.

IMPORTANT:

If you update any KeyFiles in the system, be sure to run `wm_backup` to make a new backup file and copy the files in `/usr/local/WinMagic/var` to a secure location off the machine you are working on.

Listing Key Files in the System

To list the keyfiles in the system, run:

```
# /usr/local/WinMagic/bin/wm_secdoc_ctrl dbl <major> <minor>
```

For example:

```
# /usr/local/WinMagic/bin/wm_secdoc_ctrl dbl 8 0
01 Status=84 Length=588
```

Note the index used "01" in this case for the other commands in this section.

Exporting a Key File

To export a keyfile from the system, run:

```
# /usr/local/WinMagic/bin/wm_secdoc_ctrl dbe <major> <minor>
<index> <filename>
```

For example, to export DBK from index 1 to kf1.dbk:

```
# /usr/local/WinMagic/bin/wm_secdoc_ctrl dbe 8 0 1 kf1.dbk
```

You can now take the keyfile to SecureDoc Windows or SES and change the password or add/remove encryption keys, etc.

Importing a Key File

To import a keyfile to the system, run:

```
# /usr/local/WinMagic/bin/wm_secdoc_ctrl dbi <major> <minor>
<index> <filename>
```

For example, to import DBK kf1.dbk to index 2:

```
# /usr/local/WinMagic/bin/wm_secdoc_ctrl dbi 8 0 2 kf1.dbk
```

You can import a keyfile over top of an existing keyfile. Just be careful not to overwrite the default keyfile 1 with a keyfile that does not contain the same encryption key for the HD or else an unbootable system will occur.

Deleting a Key File

To delete a keyfile from the system, run:

```
# /usr/local/WinMagic/bin/wm_secdoc_ctrl dbd <major> <minor>
<index>
```

For example, to delete DBK from index 2:

```
# /usr/local/WinMagic/bin/wm_secdoc_ctrl dbd 8 0 2
```

Using Encrypted Removable Media

If you have removable media (e.g. a USB memory stick) which has been encrypted with SecureDoc Windows, then you can mount that USB device in SecureDoc Linux and read and write to it, as long as you have the proper key in your keyfile.

NOTE: At this time SecureDoc Linux cannot encrypt removable media itself.

Mounting a USB Stick

To mount an encrypted USB stick:

1. Insert the USB stick into the PC.
2. To determine the major and minor number of the USB stick, enter:
   `# ls –l /dev/sd*` (on some systems it is "hd*")
   ```
   brw-r----- 1 root disk 8,  0 2009-04-17 06:44 /dev/sda
   brw-r----- 1 root disk 8,  1 2009-04-17 06:44 /dev/sda1
   brw-r----- 1 root disk 8,  2 2009-04-17 06:44 /dev/sda2
   brw-r----- 1 root disk 8,  3 2009-04-17 10:44 /dev/sda3
   brw-r----- 1 root disk 8, 16 2009-04-17 11:15 /dev/sdb
   ```
   Note the major number 8 and minor number 16 for the USB stick./dev/sdb.
3. Run:
   `# /usr/local/WinMagic/bin/wm_secdoc_ctrl add <major> <minor>`
   For example:
   ```
   # /usr/local/WinMagic/bin/wm_secdoc_ctrl add 8 16
   secdoc: 1239983113 DEBUG: main(): add dev=8388624
   ```
4. Run:
   ```
   # mkdir /mnt/USB
   # mount /dev/sdb /mnt/USB
   # ls /mnt/USB
   ```
   You should see your files on your USB stick.

Unmounting a USB Stick

To unmount the encrypted USB stick:

1. Run:
   `# umount /mnt/USB`
2. Run:
   `# /usr/local/WinMagic/bin/wm_secdoc_ctrl rem <major> <minor>`
   For example:
   ```
   # /usr/local/WinMagic/bin/wm_secdoc_ctrl rem 8 16
   secdoc: 1239983113 DEBUG: main(): add dev=800010
   ```
3. Remove the USB stick from the PC.

Emergency Disk

In the rare case that your system becomes unbootable, you may be able to use the Emergency Disk information you backed up off the machine in a secure location at various stages of installing SecureDoc Linux, to recover it. This information must be up-to-date, i.e., each time you encrypt/decrypt or change a keyfile you must make a new backup.

IMPORTANT:
You should contact WinMagic Support for assistance before using any of these tools.

Removing BootLogon

If your system is not yet encrypted but there is a problem with Boot Logon not passing, you can remove BootLogon as follows:

1. Turn on your PC and insert your Linux install CD/DVD and boot from it. If necessary enter the BIOS settings and make sure your PC is set to boot from CD/DVD first, before the HD.
2. Select to boot to the "Recovery mode" of your Linux distribution. What is required is to get a Linux shell and be able to access the HD in your system.
3. Once booted and at a shell, run
   # ls –l /dev/sd* (on some systems it is "hd*")
   ```
   brw-r----- 1 root disk 8,  0 2009-04-17 06:44 /dev/sda
   brw-r----- 1 root disk 8,  1 2009-04-17 06:44 /dev/sda1
   brw-r----- 1 root disk 8,  2 2009-04-17 06:44 /dev/sda2
   brw-r----- 1 root disk 8,  3 2009-04-17 10:44 /dev/sda3
   ```
   Identify the HD that is the one you installed SecureDoc on, in this case /dev/sda.
4. Transfer the Emergency Disk files you previously backed up to this system. The easiest way to do this is to copy them to a USB memory stick and insert it into your PC now. Then run something like:
   ```
   mkdir /opt
   mount /dev/sdb /opt
   ls /opt
   ```
5. Find the wm_RemoveBL script and run it as follows:
   ```
   # wm_RemoveBL --disk=/dev/sda
   ****************************************************************
   ****************************************************************

   WARNING: If you remove BootLogon from a drive that is encrypted,
   your system will be unbootable!

   ****************************************************************
   ****************************************************************
   Remove BootLogon from: /dev/sda (yes/no)?
   ```

NOTE: If you have more than one MBR backup you can specify the file to restore with
`# wm_RemoveBL --disk=/dev/sda --mbr=mbr.pre.20090623120000`

6. Enter `yes` and press ENTER to confirm the choice.
7. Reboot and remove the Linux CD/DVD. Your system should now boot as normal.
8. Follow the uninstall procedures to remove the rest of SecureDoc Linux.

Restoring SecureDoc Space

If your system is encrypted but something happens to cause it to not boot, it may be possible to recover the SecureDoc Space to correct the issue. You should contact WinMagic Support before using this tool.

To recover SecureDoc Space:

1. Turn on your PC and insert your Linux install CD/DVD and boot from it. If necessary enter the BIOS settings and make sure your PC is set to boot from CD/DVD first, before the HD.
2. Select to boot to the "Recovery mode" of your Linux distribution. What is required is to get a Linux shell and be able to access the HD in your system.
3. Once booted and at a shell, run
   # ls –l /dev/sd* (on some systems it is "hd*")
   ```
   brw-r----- 1 root disk 8,  0 2009-04-17 06:44 /dev/sda
   brw-r----- 1 root disk 8,  1 2009-04-17 06:44 /dev/sda1
   brw-r----- 1 root disk 8,  2 2009-04-17 06:44 /dev/sda2
   brw-r----- 1 root disk 8,  3 2009-04-17 10:44 /dev/sda3
   ```
   Identify the HD that is the one you installed SecureDoc on, in this case /dev/sda.
4. Transfer the Emergency Disk files you previously backed up to this system. The easiest way to do this is to copy them to a USB memory stick and insert it into your PC now. Then run something like:
   ```
   mkdir /opt
   mount /dev/sdb1 /opt
   ls /opt
   ```
5. Find the wm_SDEmgRec script and run it as follows:
   ```
   # wm_SDEmgRec --disk=/dev/sda --sdspace=SDSpace1.DAT
   ****************************************************************
   ****************************************************************

   WARNING: Restoring the SecureDoc Space may result in an
   unbootable system if done incorrectly.  You must have a current
   backup of the SecureDoc Space taken from wm_backup in SecureDoc
   Linux.  You should not continue if the disk conversion was
   interrupted or you do not have a current backup.
   ```

```
We recommend you talk to WinMagic Support before using this
utility

*****************************************************************
*****************************************************************
Restore SecureDoc Space file SDSpace1.DAT
to /dev/sda (sec 395293) (yes/no)?
```

6. Enter `yes` and press enter to confirm the choice.
7. Reboot and remove the Linux CD/DVD.

Chapter 3: Reference

Directory Structure

All directories have `-rwx------ root root` rights.

/usr/local/WinMagic

|-bin

|-boot

|-etc

|-lib

|-share

|-var

Directory	Contents
bin	• Bootlogon binary to install SD space and boot login tools • wm_boot script to create the entries in /dev directory for our kernel module • wm_bootinstall script that act as a wrapper for bootlogon binary • wm_encrypt and wm_decrypt scripts that act as a wrapper for wm_secdoc_ctrl • wm_moduleinstall and wm_moduleunistall scripts for kernel module installation • wm_secdoc_ctrl binary to start encryption/decryption process and kernel module control • wm_mbrestore will restore the MBR of the boot disk after Boot Logon is installed. The wm_bootinstall, wm_encrypt and wm_decrypt have symbolic links in the /usr/bin directory so these can be run without typing the full path.
boot	All the pre-boot binaries necessary to read SD space, hook the int13 and initial decryption: bkgd.bin, chkboot.dat, extcode.bin, h1.bin, h3.bin, h5.bin, l0.ovl, l2.ovl, mbrcode.bin, radio.bin, sdlogo.bin, boot_msg.txt, e0.bin, font.bin, h2.bin, h4.bin, hands.bin, l1.ovl, l3.ovl, menu.bin and radio_s.bin.
etc	Contains installation and program settings and temporary files from SES for SES Managed Installs.

Directory	Contents
lib	Kernel modules as files using the wm_secdoc.ko-{kernel_version}-{processor} pattern. Location for new kernel modules as a patching process or as a default installation because the wm_moduleinstall searches this directory for a suitable kernel module. The search is done using {kernel_version}-{processor] as key.
share	Contains this User Manual in PDF format as well as the License agreements and release notes.
var	Used by log files and MBR saved files. In a fresh installation this directory is empty, but after the Boot Logon installation and conversion at least four files should be there: • boot.log and encrypt.log will contain logs from both scripts and binaries • mbr.pre.{timestamp} and mbr.fin.{timestamp} contain the MBR sector before and after bootlogon installation. These can be used to restore the system and must be saved.

Interpreting Log Files

If Installation Check Fails

If `lsmod|grep wm_secdoc` returns nothing, check the `/usr/local/WinMagic/var/boot.log` file.

If No Log File

If the file does not exist, you did not start the bootlogon installation sequence: try it again (see "Installing Boot Logon" on page 7).

If Log File Contains Evident Errors

If the log file exists and contains errors, the next step depends on the error message.

If Log File Contains No Evident Errors

If no evident errors are found in the `boot.log` file, check the `/usr/local/WinMagic/var/startup.log` file. This is the log file of the `wm_boot` script that checks for kernel module and creates the device link. To manually check the kernel version and the module version, enter:

```
# uname –r
2.6.25.5-1.1-pae
```

followed by:

```
# ls -l /usr/local/WinMagic/lib

total 13428

-rw-r--r-- 1 root root 2527941 2009-04-14 10:09 wm_secdoc.ko-2.6.18-128.el5-
i686

-rw-r--r-- 1 root root  259405 2009-04-14 10:09 wm_secdoc.ko-2.6.18.2-34-
default-i686

-rw-r--r-- 1 root root  277942 2009-04-14 10:09 wm_secdoc.ko-2.6.25.5-1.1-
pae-i686

-rw-r--r-- 1 root root 3454731 2009-04-14 10:09 wm_secdoc.ko-2.6.27.19-
170.2.35.fc10.i686-i686

-rw-r--r-- 1 root root 3476863 2009-04-14 10:09 wm_secdoc.ko-2.6.27.19-
170.2.35.fc10.i686.PAE-i686

-rw-r--r-- 1 root root 3452987 2009-04-14 10:09 wm_secdoc.ko-2.6.27.5-
117.fc10.i686-i686

-rw-r--r-- 1 root root  256387 2009-04-14 10:09 wm_secdoc.ko-2.6.27.7-9-pae-
i686
```

A kernel version must match one of the SecureDoc module names. If it does not, contact WinMagic Technical Support. If it does, follow the process below:

1. Enter:

   ```
   # insmod /usr/local/WinMagic/lib/wm_secdoc.ko-{kernel_version}-{processor}
   load_probe=1
   ```

2. If the result is as shown below, contact WinMagic Technical Support.

   ```
   insmod: error inserting 'wm_secdoc.ko-{kernel_version}-{processor}': -1 Invalid module
   format
   ```

   If no errors are returned, repeat `# lsmod|grep wm_secdoc`.

3. If this returns nothing, unload the module and install it manually:

   ```
   # rmmod wm_secdoc

   # cp /usr/local/WinMagic/lib/wm_secdoc.ko.{kernel_version}
   /lib/modules/{kernel_version}/kernel/crypto

   # depmod -a

   # mkinitrd

   # reboot
   ```

4. Wait until the machine reboots and repeat `# lsmod|grep wm_secdoc`. If errors persist, try one of the additional checks below or contact WinMagic Technical Support.

Addition Check

1. Enter:
   # ls -la /dev/wm_secdoc
   crw-r--r-- 1 root root 254, 0 2008-03-06 10:05 /dev/wm_secdoc
   # cat /proc/devices
   Character devices:
   ...
   180 usb
   189 usb_device
   254 wm_secdoc

   Block devices:
   ...
   135 sd
   253 device-mapper
   254 mdp

   This indicates that the module is loaded with the 254 char major device number and no other module has this number; also that the link in the `/dev` directory is created correctly.
2. If the output of # ls -la /dev/wm_secdoc is an error, check the `/etc/init.d/boot.local` file and be sure it contains:
   #cat /etc/init.d/boot.local
   ...
   . /usr/local/WinMagic/bin/wm_boot
   ...

   If the line is not there, something was wrong with module installation. If the bootlogon is correctly installed then add this line:
   # echo ". /usr/local/WinMagic/bin/wm_boot" >> /etc/init.d/boot.local
   #reboot

   After reboot , repeat all checks from the start (`lsmod|grep wm_secdoc).`

Checking Mounted Partitions

1. At any time, check the mounted partitions with:

```
# mount
/dev/sda5 on / type reiserfs (rw,acl,user_xattr)
proc on /proc type proc (rw)
sysfs on /sys type sysfs (rw)
debugfs on /sys/kernel/debug type debugfs (rw)
udev on /dev type tmpfs (rw)
devpts on /dev/pts type devpts (rw,mode=0620,gid=5)
/dev/sda6 on /opt type reiserfs (rw)
/dev/sda9 on /extra type reiserfs (rw)
securityfs on /sys/kernel/security type securityfs (rw)
none on /proc/fs/vmblock/mountPoint type vmblock (rw)
```

2. Because the `/partitions` is mounted on `/dev/sda5`, check the `/dev/sda` device (the `/dev/sda` is used for SCSI/SATA disks and `/dev/hda` is used by IDE disks):

```
# parted -s /dev/sda unit s print
Model: ATA Maxtor 6Y080L0 (scsi)
Disk /dev/sda: 160086528s
Sector size (logical/physical): 512B/512B
Partition Table: msdos

Number  Start       End         Size       Type      File system  Flags
 1      63s         64259s      64197s     primary   fat16        , , , , , , , , , type=de, ,
 2      64260s      61769924s   61705665s  primary   ntfs         boot, , , , , , , , , type=07, ,
 3      61769925s   160071659s  98301735s  extended              , , , , , , lba, , , type=0f, ,
 5      61769988s   82268865s   20498878s  logical   reiserfs     , , , , , , , , , type=83, ,
 8      82268928s   84373379s   2104452s   logical               , , , , , , , , , type=82, ,
 9      84373443s   123202484s  38829042s  logical   reiserfs     , , , , , , , , , type=83, ,
 6      123202548s  152890604s  29688057s  logical   reiserfs     , , , , , , , , , type=83, ,
 7      152890668s  160071659s  7180992s   logical   ext2         , , , , , , , , , type=83, ,
```

3. From this output you can see that the boot flag is present, the `/partition` is the index 5 (/dev/sda5) and the type=83. Also, the swap partition is the one with type=82 and id 8. Now it is possible to calculate if the SD space is already created, making the difference between the start sector of the swap next partition (id 9 in our case), meaning 84373443, and the end sector of the swap partition, meaning 84373379. The difference must be greater than 12500 sectors. If the swap partition is the last on the disk, then the difference is between the last sector of the disk, meaning 160086528, and the last sector of the swap partition.
4. Further, check the block devices present in system:

```
# ls -la /sys/block
total 0
drwxr-xr-x 13 root root 0 Mar  6 03:24 .
drwxr-xr-x 11 root root 0 Mar  6 03:24 ..
drwxr-xr-x  5 root root 0 Mar  6 08:25 fd0
drwxr-xr-x  6 root root 0 Mar  6 08:25 hda
drwxr-xr-x  4 root root 0 Mar  6 08:24 loop0
drwxr-xr-x  4 root root 0 Mar  6 08:24 loop1
drwxr-xr-x  4 root root 0 Mar  6 08:24 loop2
drwxr-xr-x  4 root root 0 Mar  6 08:24 loop3
drwxr-xr-x  4 root root 0 Mar  6 08:24 loop4
drwxr-xr-x  4 root root 0 Mar  6 08:24 loop5
drwxr-xr-x  4 root root 0 Mar  6 08:24 loop6
drwxr-xr-x  4 root root 0 Mar  6 08:24 loop7
drwxr-xr-x 13 root root 0 Mar  6 14:18 sda
drwxr-xr-x  5 root root 0 Mar  6 08:25 sr0
```

5. If both the hda and sda devices are present in the system, check:

```
# parted -s /dev/hda unit s print
```

The most important check is the content of the GRUB/LILO files: `/boot/grub/device.map` and `/etc/lilo.conf`. Typically, only one of these files is present in the system. For example, for a GRUB file the content can be:

```
# cat /boot/grub/device.map
(fd0)   /dev/fd0
(hd0)   /dev/sda
```

This shows that the boot disk is /dev/sda so a comparison can be made to see if the / and swap partitions are part of this disk.